

Automatic License Plate Recognition System

Saprem Dalal
Aaron D'Souza



Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769 008, Orissa, India

Automatic License Plate Recognition System

A thesis submitted on

9th May, 2011

in partial fulfillment of the requirements for the degree of

Bachelor of Technology

in

Computer Science and Engineering

by

Saprem Dalal

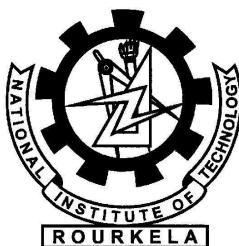
(Roll no. 107CS030)

Aaron D'Souza

(Roll no. 107CS040)

Under the guidance of

Prof. Dr. Banshidhar Majhi



Department of Computer Science and Engineering

National Institute of Technology Rourkela

Rourkela-769 008, Orissa, India

National Institute of Technology Rourkela



Certificate

This is to certify that the project entitled, ‘**Automatic License Plate Recognition System**’ submitted by **Saprem Dalal and Aaron D’Souza** is an authentic work carried out by them under my supervision and guidance for the partial fulfillment of the requirements for the award of **Bachelor of Technology Degree in Computer Science and Engineering** at **National Institute of Technology, Rourkela**.

To the best of my knowledge, the matter embodied in the project has not been submitted to any other University / Institute for the award of any Degree or Diploma.

Date - 09/05/2011

Rourkela

(Prof. Dr. B. Majhi)

Department of Computer Science and Engineering

Acknowledgments

We express our profound gratitude and indebtedness to **Prof. Dr. B. Majhi** , Department of Computer Science and Engineering, NIT Rourkela for introducing the present topic and for his inspiring intellectual guidance, constructive criticism and valuable suggestion throughout the project work.

We are also thankful to **Prof. Dr P.K. Sa**, Department of Computer Science and Engineering for guiding and motivating us throughout the project.

Finally, we would like to thank our parents and colleagues for their support and motivation to complete this project.

Date - 09/05/2011

Rourkela

Saprem Dalal
Aaron D Souza

Abstract

Intelligence surveillance is an important commodity in traffic-based systems. Automatic License Plate Recognition (ALPR) is a challenging area of research. This work deals with problems related to artificial intelligence, neural networks and machine vision in the construction of an automatic license plate recognition (ALPR) system. This is done using mathematical principles and algorithms. These intelligent systems help in traffic monitoring during rush hours, road safety, commercial applications like in car parking lots and law enforcement. In this paper, a license plate recognition system is proposed which uses captured digital images of the rear or front of a vehicle and can be easily applied to commercial car park systems for access to parking spaces and also to prevent car theft issues.

Contents

1	Introduction	4
1.1	Problem Statement	4
1.2	Challenges	4
1.3	Types of License plate recognition system	5
1.4	Challenges in ALPR	5
2	An Overview of the System	9
2.1	Preprocessing Techniques for localization of the license plate	9
2.2	Localization of Licence Plate	11
2.2.1	Edge Detection	11
2.2.2	Sobel filter	12
2.2.3	Connected components	15
2.2.4	Bounding box	16
2.2.5	Selecting the Best Bounding Box	16
2.2.6	Cropping the Bounding Box	18
2.3	Principles of Character Segmentation	18
2.3.1	Preprocessing stage	19
2.3.2	Segmentation of plate using a horizontal projection	21
2.4	The Image Deskewing Mechanism	25
3	Optical Character Recognition	27
3.1	Introduction	27
4	Applications	30
5	Results	30
6	Conclusion	34
7	References	35

1 Introduction

1.1 Problem Statement

To design an Automated Licence Plate Recognition System for the application in car parking.

1.2 Challenges

Generally, an automatic license plate localization and recognition (ALPR) system is made up of three modules; license plate localization, character segmentation and optical character recognition modules (Fig.1)

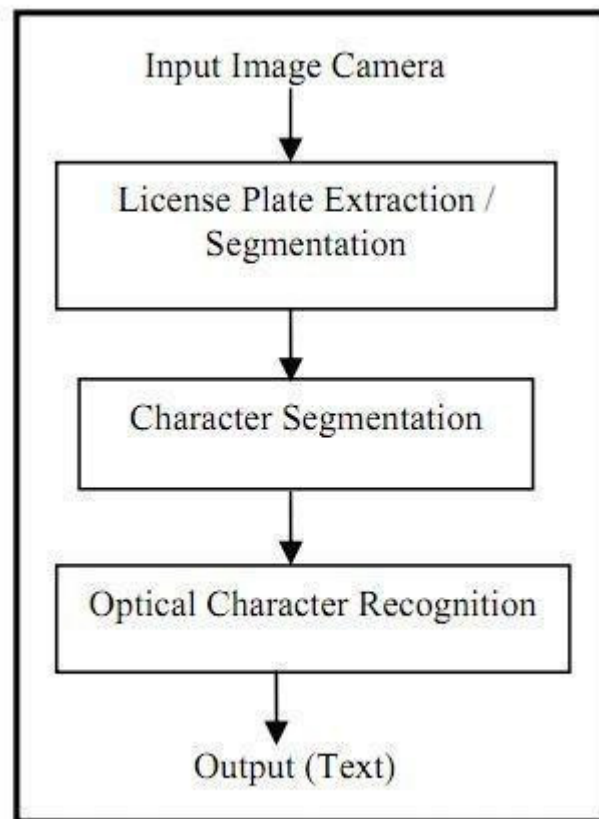


Fig. 1. Flowchart of a Typical ALPR System

- Firstly, license plate localization from vehicle images;
- Secondly, character segmentation from localized license plate, and

- Finally, optical character recognition of extracted characters.

The most common solutions to license plate localization in digital images are through the implementation of edge extraction, morphological operators and Sobel operator. An edge approach is normally simple and fast. Sobel operator for edge detection gives positive effects on image. The localization of license plates via morphological based approaches is not susceptible to noise but is very slow in execution.[3]

After the localization of the license plate comes the character segmentation process. Common character segmentation processes are based on histogram analysis and thresholding. Other recent approaches proposed are the use of artificial neural networks.

The final stage of an ALPR system is the character recognition process. To deal with a number of variations found in characters across different license plates will require the segmented character to undergo some preprocessing steps such as normalization and skew correction. These additional steps prove to be beneficial as it greatly reduces the required computation time[2].

1.3 Types of License plate recognition system

1. Online ALPR system: In an online ALPR system, the localization and interpretation of license plates take place instantaneously from the incoming video frames, enabling real-time tracking through the surveillance camera.
2. Offline ALPR system: An offline ALPR system, in contrast, captures the vehicle images and stores them in a centralized data server for further processing, i.e. , for interpretation of vehicle license plates.

1.4 Challenges in ALPR

In the developed countries the attributes of the license plates are strictly maintained. For example, the size of the plate, color of the plate, font face / size / color of each character, spacing between subsequent characters, the number of lines in the license plate, script

etc. are maintained very specifically. Some of the images of standard license plates, used in developed countries, are shown in the figures 2 and 3[8].



However, in India, the license plates are not yet standardized across different states, making localization and subsequent recognition of license plates extremely difficult. Moreover, in India license plates are often written in multiple scripts. Below are some of the typical Indian license plates with variations in shape, size, script etc.[8]



Automatic license plate recognition has two major technological requirements

1. The quality of the license plate recognition algorithms.
2. The quality of the image acquisition(camera and the illumination conditions).

The better the algorithms are:

1. Higher is the recognition accuracy.
2. Faster is the processing speed.

3. Wider is the range of picture quality it can be used on.

Generally one LPR software can read plates from one particular country only. This is because the geometrical structure of the plate and the orientation, fonts and syntax were important parts of the LPR system. Without the prior knowledge of the plate geometry (character distribution, character spacing, plate colour, dimension ratios etc.), the algorithm may not even find the plate in the captured image.

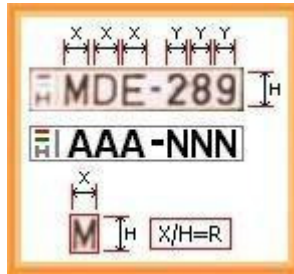


Plate geometry and basic syntax (<http://www.platerecognition.info/1102.htm>)

Furthermore there are a wide variety of plate types:

1. Black characters on white/light background.
2. White characters on black/dark background.
3. Single row plates.
4. Multi row plates.

If a LPR system cannot utilize such useful information like the plate structure, it loses a helpful aid derived from its input data. This could result in the reduction of the license plate recognition system accuracy. Without using prior information of the plate, the remaining part of the recognition system should be significantly more robust and this leads to more challenges.

The image acquisition technique determines the captured image quality of the license plate with which the detection algorithms have to work. Better the quality of the acquired images, higher is the accuracy one can achieve.

A good captured image has the following properties:

1. Good spatial resolution,
2. Good sharpness,
3. high contrast,
4. Adequate lighting conditions,
5. decent angle of view.

Here are some problematic images obtained from (<http://www.platerecognition.info/1102.htm>)



Low spatial resolution (too small characters on the plate)



Blurred image



Low contrast



Overexposure



Bad lighting conditions (shadow and strong light)



High distortion

SOURCE: <http://www.platerecognition.info/1102.htm>

2 An Overview of the System

2.1 Preprocessing Techniques for localization of the license plate

The preprocessing techniques mentioned below are implemented in the present work in order to address the issues mentioned above

1. Conversion to Gray Scale: The Red, Green and Blue components are separated from the 24-bit colour value of each pixel (i,j) to calculate the 8-bit gray value using the formula[7]:

$$\text{gray}(i,j)=0.59*R(i,j)+0.30*G(i,j)+0.11*B(i,j)$$

2. Median filtering: The Median filter is a non-linear filter, which is used to replace the gray value of each pixel with the median of the gray values of its neighbors. We have used 3 3 masks which helps in getting eight neighbors of the pixel and their corresponding gray values. This operation helps in removing salt-and-pepper noise from image
3. Enhancing the contrast: Contrast of each image can be enhanced through the histogram equalization technique. There are a total of 256 numbers of gray levels (0 to 255) that are used for stretching the contrast. Let the total no. of pixels in an image be n and the no. of pixels having gray level k let it be nk. Then the probability that a gray level k will be occurred is, $P_k = n_k / N$. The stretched gray level S_k is calculated using the formula given below . It uses the cumulative frequency of occurrence of a particular gray level k in the image given:

$$S_k = \sum_{j=0}^k \frac{n_j}{N} \times 255$$

Here, 255 is the maximum gray level present in the enhanced image Pictures below are the Screenshots of the preprocessing :



Figure 1.

(a) Grayscale Image

(b) Noise Reduced Image



(c) Contrast Enhanced Image

2.2 Localization of Licence Plate

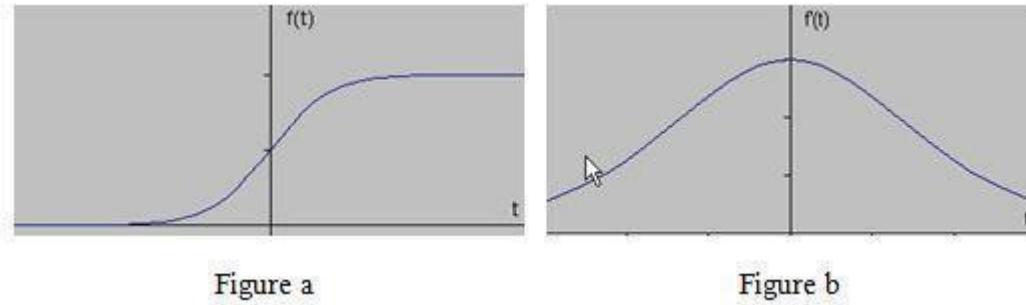
This technique is used to identify the potential licence plate region from the given image . The main objective of such kind of systems are to loacalize the lincence plate region from the images of the vehicles that are captured from the roadside camera. Quality of the image forms an important part of the this technique so preprocessing the image helps in improving the quality.

Number plates usually appear to have high contrast areas in the image (black-and-yellow or black-and-white). The numbers and letters are placed in the same row (i.e. at an identical vertical level), which results in frequent changes in intensity horizontally. This provides the basis for detecting the changes in the horizontal intensity, as the rows that will contain the number plate are expected to show sharp variations. The reason for this sharp variation is the contrast between the letters and its background.

2.2.1 Edge Detection

Edges helps to characterize the boundaries and therefore are a problem of fundamental importance while processing the image . Edges in images are the areas where strong intensity contrasts are present, a sudden variation in the intensity from one pixel to the next. Detecting the edges of an image significantly reduces the amount of data and it helps in filtering out the useless information, while preserving the important structural properties of an image. There are many ways to perform the edge detection. However, the majority of various methods can be grouped into two different categories, gradient and Laplacian. The gradient methods detect the edges by finding out the maximum and minimum in the first derivative of the image. The Laplacian method searches for the zero crossings in the second derivative of the image to find the edges. An edge has the one dimensional shape of the ramp and calculating the derivative of the image can highlight its location. Suppose we have the signal shown below, with an edge shown by the sudden variation in intensity below in Figure a[11]. If we take the gradient of this given signal (which, in one dimension, is the first derivative with respect to t) we get the following

shown result in Figure b



As shown, the derivative shows maximum located at the center of the edge in the original signal. Such a method of finding an edge is characteristic of the gradient filter family of the edge detection filters and it includes the Sobel method. A pixel location is declared as an edge location if the value of the gradient is exceeding some threshold. As discussed before, the edges will have higher pixel intensity values than their surrounding pixels. So once a threshold has been set, you can compare the gradient value to threshold value and an edge can be detected whenever the threshold is exceeded. Also, when the first derivative is at an maximum, the second derivative turns out to be zero. As a result, another alternative which can be used to find the location of an edge is to find the zeros in the second derivative. This method is called as the Laplacian and the second derivative of the signal is as shown below:

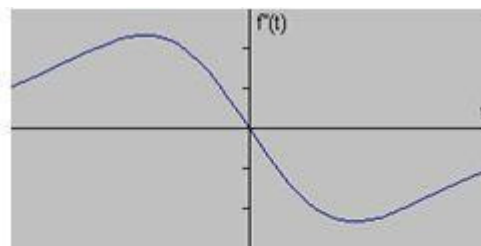


Figure c

2.2.2 Sobel filter

The theory of the one-dimensional analysis can be carried over to two-dimensions as long as there is an accurate approximation for calculating the derivative of a two-dimensional

image[4]. The Sobel operator helps to perform a 2 Dimensional spatial gradient measurement on an image. Generally it is used to find the approximate absolute gradient magnitude at each point on an input grayscale image. The Sobel edge detector make use of a pair of 3x3 convolution masks, one estimates the gradient in the x-direction (columns) and the other estimates the gradient in the y-direction (rows). A convolution mask is usually very much smaller than the actual image. Therefore the mask is slid over the image, and it manipulates a square of pixels at a time. The actual Sobel masks are shown below[11]:

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

The magnitude of the gradient is calculated by using the formula:

$$|G| = \sqrt{Gx^2 + Gy^2}$$

An approximate magnitude can be calculated using:

$$|G| = |Gx| + |Gy|$$

Here in this work, we have extracted the edges created by the characters within the license plate. Here the Sobel edge operator is used for the detection of edge gradients. It is observed that when the characters in the license number are written horizontally the vertical edges of each character do appear at regular interval and they tend to have a specific height. The pattern and the concentration of the vertical edges also remain in conformity with the pattern seen of the license number. Such appearance of vertical edge pattern is statistically seen to occur mostly within the license plate, other than that no where else it is observed within the natural scene of the image[5]. In this present

work, we have explored this kind of phenomenon to find the license plate region inside the image. The vertical edge at point (x,y) can be found using the following formula:

$$gradV(y, x) = \sqrt{\left(\sum_{n=-1}^{+1} \sum_{m=-1}^{+1} V_mask(n, m) img_contrast(y + n, x + m) / 4 \right)^2}$$

where, the term img contrast is the enhanced image over which edge detection algorithm is operated, the term V mask is the Sobel's mask used for vertical edge detection as given below and the term gradV is called the vertical edge gradient.

$$V_mask = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

Depending upon the value of gradV we have binarized the edge of the image using the threshold gradV (mean edge gradient value) and have formed the edge image img edge. The result of edge detection after binarization is applied on it is shown in figure below

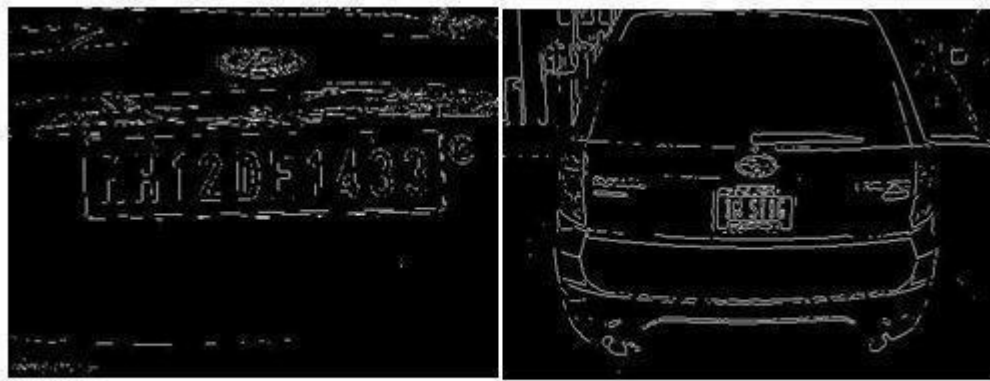


Figure 2 : After applying Sobel filter and binarizing the image

Number plates usually appear as high contrast areas in the image (black-and-white or black-and-yellow). The letters and numbers are placed in the same row (i.e. at identical vertical levels), resulting in frequent changes in the horizontal intensity. This provides the reason for detecting the horizontal changes of the intensity, since the rows that contain the number plate are expected to exhibit many sharp variations.

The algorithm first determines the extent of intensity variation for each row. While in the second step it selects the adjacent rows which exhibit the biggest changes. Number plates are highly probable to be present in these rows. The horizontal position of the number plate must also be determined, which is done by using the previously determined values that characterize the changes. In the similar way the vertical position of the license plate is determined. The variations are the highest at the letters (black letters on white background); therefore this is where the rate of change within a row is expected to be the highest.

2.2.3 Connected components

1. 4-connected: 4-Connected pixels are the neighbors to every pixel that touches on one of their edges. These pixels are connected both horizontally and vertically. In the terms of pixel coordinates, every pixel having the coordinates or is connected to the pixel at (x,y) [12].
2. 8-connected: 8-connected pixels are the neighbors to every pixel that touches on one of their edges or the corners. These pixels can be connected horizontally, vertically, and diagonally. In addition to the 4-Connected pixels, each pixel with the coordinates or is connected to pixel at (x,y) [12].

The connected Components are then found out in the filtered image and each connected component is labeled.

2.2.4 Bounding box

The minimum or the smallest bounding or the enclosing box for any point set in N dimensions is the box with the smallest measure (of area, volume, or hypervolume in higher dimensions) within which all points lie. When the other kinds of measure are used, the minimum box is usually called accordingly depending on the measure used, e.g., "minimum-perimeter bounding box". The minimum bounding box of any point set is same as the minimum bounding box of its convex hull, this is a fact which can be used heuristically to speed up computation. The term "box"/"hyperrectangle" has come from its usage in Cartesian coordinate system, where it can be indeed visualized as a rectangle (in two-dimensional case), rectangular parallelepiped (in case of three-dimensional), etc. In the case of two-dimensional it is called the minimum bounding rectangle[13].

In other words it is a rectangle which has the minimum height and width that covers all the pixels present in a particular connected component or region. The following Figure 3 shows the bounding boxes on the filtered image.

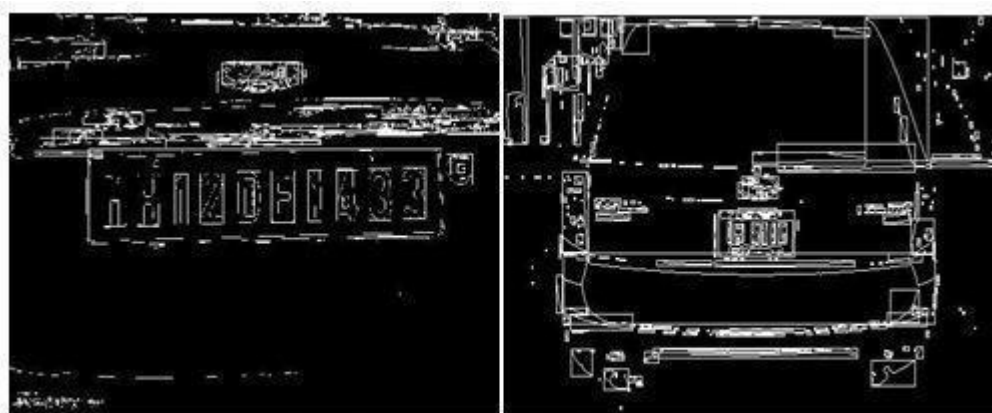


Figure 3 : Bounding boxes in the image

2.2.5 Selecting the Best Bounding Box

In a figure there can be many bounding boxes identified but to select the best possible candidate for licence plate region requires to identify few properties of the licence plates as discussed below:

1. Contrast present in the bounding box: As the license plate consists of dark colored

numbers to the lighter background or vice versa .The centre row of the box can be scanned and the total number sudden contrast change can be recoded ,if the image is binarized.The box having the maximum changes in contrast within the box is the possible candidate for the license plate.

2. Aspect Ratio: The aspect ratio of an image is the ratio of the width of the image to its height.The inverse of the aspect ratio should be less than 1 for any licence plate .Hence , all the regions which dont not satisfy this property can be rejected.
3. Width of licence plate: The width of the licence plate region also has a threshold limit ,it cannot be more than some threshold value .In this project after analyzing various dataset we have used the threshold limit to five times the height of the licence plate so :

$$Width < (5 * Height)$$

The bounding box regions not satisfying this condition are rejected.

4. Total number of pixels in Licence plate region: Another factor that separates licence plate region from the rest is the amount of pixel that it has in it .In in project, the distance in between the camera and the vehicles plate will be constant in all the images as the camera is triggered when the vehicle hits the laser .After analysis various dataset we have come to conclusion that the minimum threshold for total number of pixels in the licence plate region is 1/100th of the total number of pixels in the image and maximum threshold as 1/100th of the total number of pixels in the image plus 1500 pixels.Bounding boxes falling outside this range are rejected .The following figure 4 shows the bounding box finally identified



Figure 4: The Bounding Box satisfying the properties of licence plate

2.2.6 Cropping the Bounding Box

After identifying the best possible bounding box candidate for the licence plate the coordinates of the bounding box are noted and the box is cropped from the image and sent to Character segmentation module for further processing as shown in figure:



Figure 5 :Shows the cropped bounding box from the image.

2.3 Principles of Character Segmentation

The character segmentation process acts as a bridge between the license plate localization and optical character recognition modules. Its main function is to segment the characters in the selected candidate region (extracted license plate) such that each character can be sent to the optical character recognition module individually for recognition.

Normalized or standardized license plates are important criteria for efficient segmentation because if numbers are of a fancy format the conditions of the license plate as described above may not hold true. This is the main reason why in India, where license plates are not normalized we do not have widespread license plate recognition systems to date. The government of India has taken an initiative in this direction and it is now a must for all vehicles to have a high security number plates which adhere to certain well defined

guidelines[1].

Once the license plate is localized we proceed to obtain the individual characters. A license plate as described above has high intensity variation regions (due to alternating white and black regions). This forms the basis for character segmentation. Sometimes it is observed that along with the license numbers, various texts may be present (indicating state names etc.), which have to be removed. By various observations we observed that for the license plate regions the amount of white (number or text) on black (background region) or vice versa, is specific for the number regions and falls within a certain range. We ignore those regions which are out of range to isolate the number regions[5]. Morphological techniques are used to remove small white areas which escape range corrections (certain shadows or text which show similar patterns to numbers). Finally individual characters are extracted to pass on through the optical character recognition (OCR) system. Segmentation is one of the most important processes in the automatic number plate recognition, because all further steps rely on it. If the segmentation fails, a character can be improperly divided into two pieces, or two characters can be wrongly merged together which would lead to the failure of following stages of recognition[8].

We can use a horizontal projection of a number plate for segmentation. If we assume only one-row plates, the segmentation is a process of finding horizontal boundaries between characters.

The second phase of the segmentation is an enhancement of segments. The segment of a plate contains besides the character also undesirable elements such as noise due to shadows or defects in camera equipment as well as redundant space on the sides of character[1]. There is a need to eliminate these elements and extract only the individual characters.

2.3.1 Preprocessing stage

Before we can proceed with the segmentation stage, we must ensure that the plate obtained is cleared off most of the unwanted characters or graphics like state name or flags etc[8]. We proceed to do so by scanning the plate vertically and horizontally and ignoring

those rows and columns which have too much white or black. This is justified as those areas containing the numbers have black areas which lie in a particular range[9]. This range by experiments was found to lie between 0.2 and 0.8 times the number of pixels horizontally and vertically.



Extracted license plate

The character segmentation process takes the extracted license plate region from the preceding module as the input. The input is a colored JPEG image. For our process we work with only binary images and thus the first part of segmentation is binarisation of the image.



Resultant Binary Image



Inverted Binary Image

The binary image thus obtained has certain unwanted areas which need not be considered during the OCR process and may even hamper the detection process. For example in the above figure we have screw marks and country code written on the plate which are of no relevance to the recognition of authentic license numbers. Thus these regions

have to be filtered out.

We use the concept of connected components to filter small areas out of the plate region. Components with pixels lower than a particular threshold are converted to background and thus ignored[1]. We thus obtain a relatively clear binarised plate region suitable for segmentation and relatively free of noise and redundant regions.



2.3.2 Segmentation of plate using a horizontal projection

We compute a horizontal projection $px(x)$ of the plate $f(x, y)$. We use this projection to determine horizontal boundaries between segmented characters[5]. These boundaries correspond to peaks in the graph of the horizontal projection. The goal of the segmentation algorithm is to find peaks, which correspond to the spaces between characters. At first, there is a need to define several important values in a graph of the horizontal projection $px(x)$.

- V_m : The maximum value contained in the horizontal projection
- V_a : The average value of horizontal projection
- Max peak : The highest peak in the segmented plate image

The aim of this step is to find the boundaries between two consecutive characters on the number plate. By obtaining the horizontal projection we can consider those peaks which are greater than a certain threshold value as these boundaries. From the above figure we consider only those peaks which are greater than 0.9 times the max peak value as legitimate boundaries and hence obtain the co-ordinates of each character[1].

We take 0.9 the max peak value to account for noise which may have escaped corrections

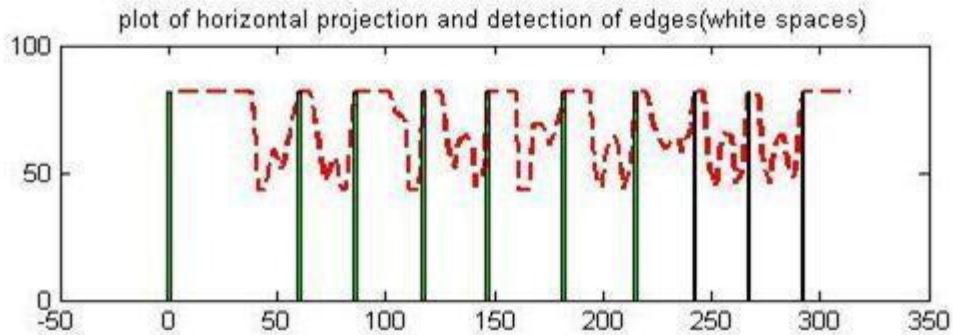


Fig.: Horizontal projection of plate with detected peaks. Detected peaks are denoted by green vertical lines.

carried out in previous steps. The regions between consecutive peaks contain one character each which can be thereafter extracted from the plate region and sent for character recognition.

The last step of the phase of character segmentation is to obtain the individual characters. This is fulfilled by using the above obtained boundary values. These individual characters are then fed to the next step of analysis which is the final step called the Optical character recognition.

Before the characters can be fed to the next step we have to remove redundant background areas from the individual images of characters. The resultant images will consist of properly bounded numbers or alphabets of the license plate characters which will make our work of OCR easier.

We do this by scanning each image first horizontally and then vertically to obtain the startrow, endrow, startcolumn and endcolumn which denote the two dimensional boundary values in which the numbers or alphabets fit perfectly[1]. Sample images are shown

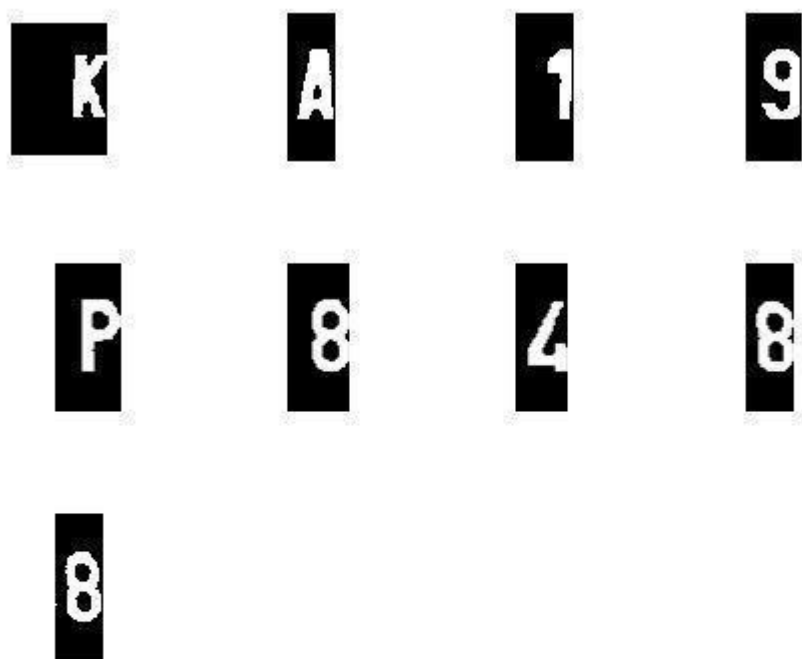


Fig.: Individual characters obtained by using peaks as indications of white spaces between two consecutive characters.

below.



Other examples of Character Segmentation:



Fig.: A] Sample localized plate



B] Binarised Image

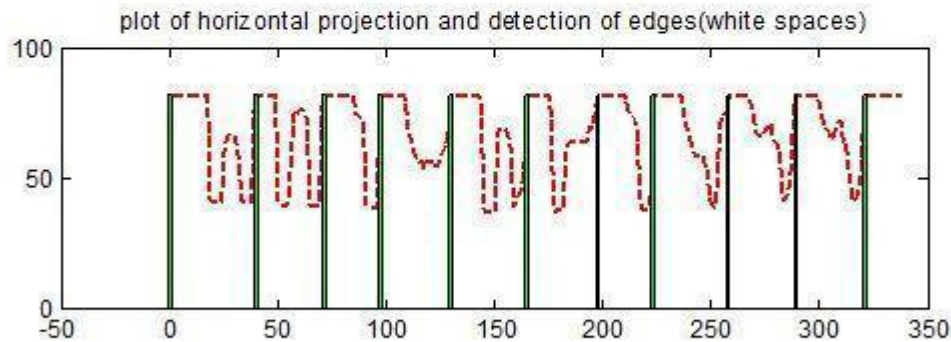


Fig.: Horizontal projection showing peaks between characters

2.4 The Image Deskewing Mechanism

Quite often we may find that the localized license plate obtained from the above steps are skewed or rotated by a certain angle[5]. This can be attributed to either:

- The camera angle during exposure
- Orientation of the vehicle with respect to the desired camera

Either way, image skew could severely hamper subsequent steps of character extraction and recognition. It is thus desired to deskew the image before passing the localized plate on to the next LPR step. Below is a sample image illustrating the need of Deskewing techniques[1].

The above skewed image could result from either of the above two reasons. If this skew is left uncorrected it could lead to the failure of segmentation and extraction steps. The below images show the resultant extracted characters as a result of the above skew[8]. It can be seen that a slightly larger skew angle would result in no defined vertical peaks or white spaces between individual characters thus rendering segmentation process ineffective.

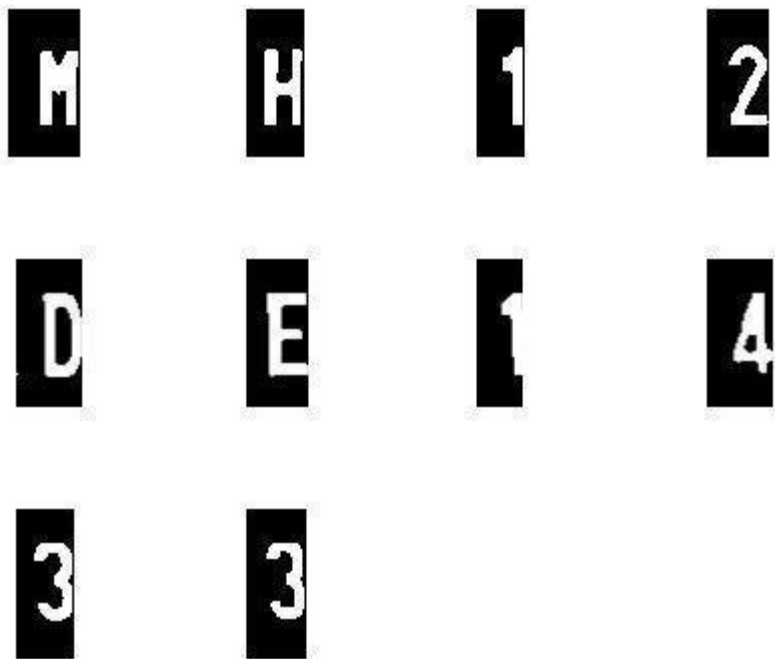


Fig.: Individual characters obtained



Skewed captured image 1



Binarised image 1

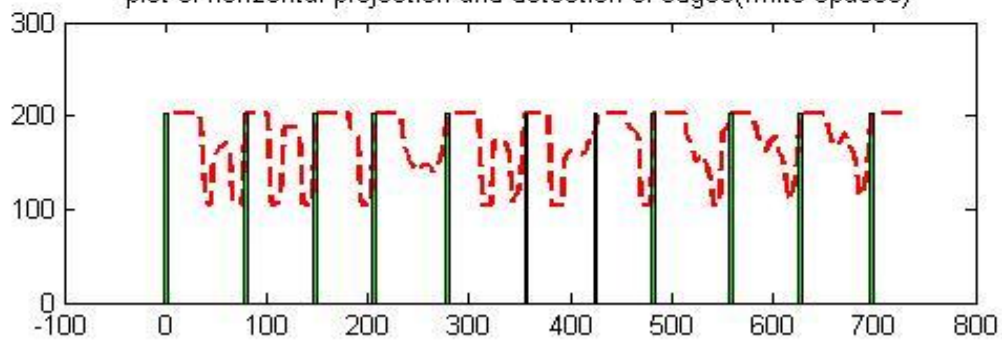
original image



deskewed image



plot of horizontal projection and detection of edges(white spaces)



3 Optical Character Recognition

3.1 Introduction

The neural networks are typically made up of many artificial neurons. An artificial neuron is an analogy to biological neuron. It is simply electronically modeled to the biological neuron. The number of neurons that are used depends on the task at hand. The number of neurons used can be few as two or three or large as two or several thousand. There are many ways of connecting artificial neurons together to form a neural network. Some of the ways are discussed below.

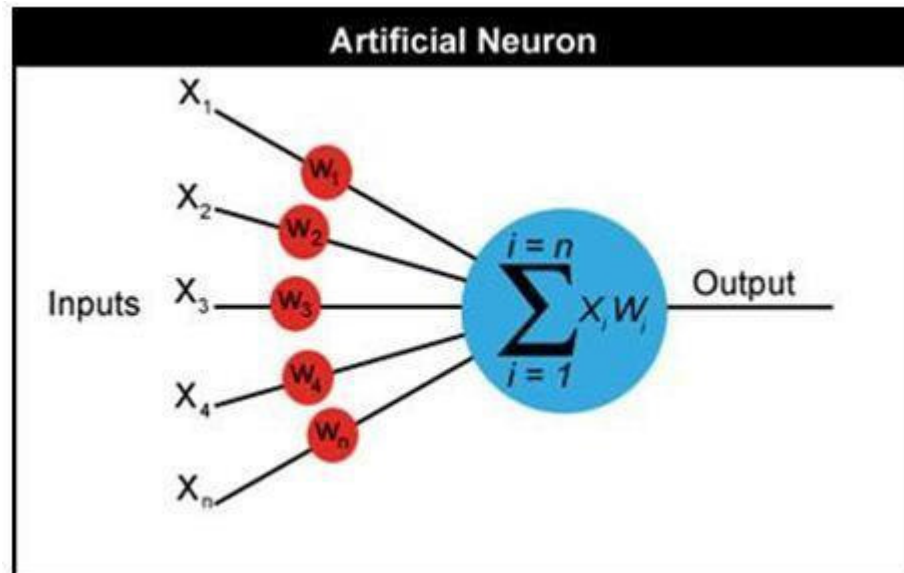
- Feedforward network:

In Feedforward Neural Network each input into the neuron has its own weight associated with it. A weight can simply be a floating point no. and it is these that we adjust when we come to train the network. The weights in most of the neural networks can be both negative and positive, therefore it helps in providing excitatory or else inhibitory influences to each input. As each of the input enters the nucleus it is then multiplied by its weight. The nucleus sums up all these new input values and gives us the activation which is again a floating point no. and can be negative or positive. The threshold value is decided and if the activation value is greater than a threshold value the neuron outputs 1 (considering there are two outcomes 1 and 0 to the input) as a signal. If the activation is less than the threshold value the neuron then outputs zero.

A neuron can take any number of inputs from one to n , here n is the total number of inputs. The inputs therefore may be represented as $x_1, x_2, x_3 \dots x_n$. The corresponding weights for the inputs can be represented as $w_1, w_2, w_3 \dots w_n$. The weighted sum of the the links and its corresponding weight is called the activation value as discussed above:

$$a = x_1w_1 + x_2w_2 + x_3w_3 \dots + x_nw_n [14]$$

Where, a is the activation value

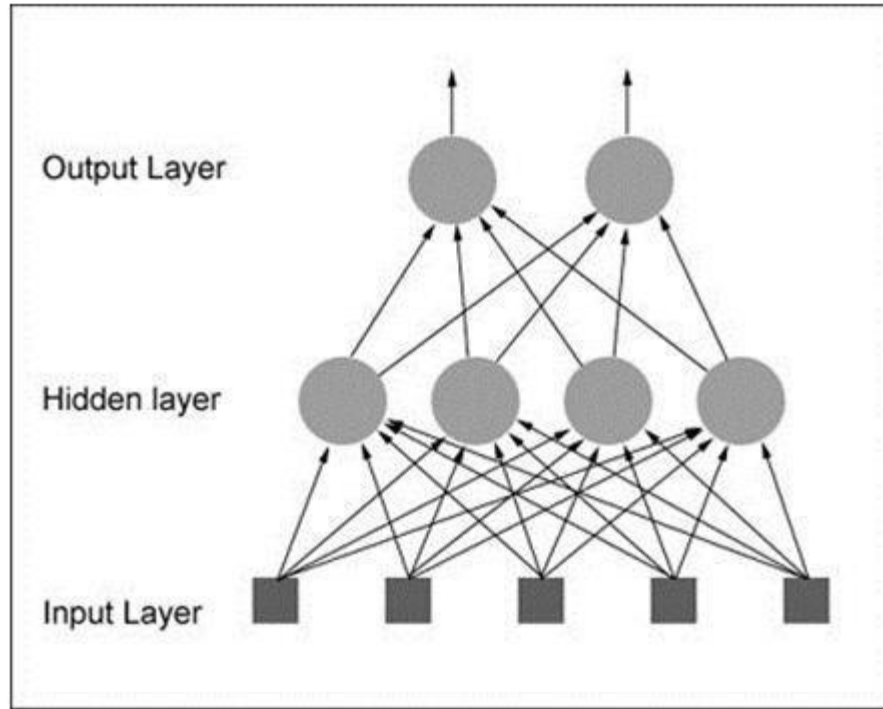


We have to link several of these neurons, one way in which it can be done is by organizing the neurons into a design called as the feedforward network . It gets its name from the way the neurons of the previous layer feed their output forward to the next layer until the final output is got from the neural network. This is how a feedforward network looks like:[14]

Each of the input is sent to every neuron of the hidden layer and each hidden layers neurons output is connected to every neuron of the next layer. There is no predefined number of neurons to be present in a particular layer it can be arbitrary and it totally depends on the problem.

- **Back Propagation Network Algorithm:** A Back Propagation networks learns by example various sets of datasets are provided as input .The various inputs provided helps the network to calculate and recalculate the networks weight value so that when the network is trained it can give the required output.

The network is initialized by first setting random weights which generally have very small valuesuch as values between -1 and 1.There are two passes in the Back Propagation algorithm .After the network is set up with the randownm weights the output is calculated this is called the forward pass.the result obtained in the for-



ward pass may not be equal to the required result or the target and so the error is calculated for each neuron which is , $\text{Target} - \text{Actual Output}$.The error calculated for each neuron is then mathematically used to change the weights so the next time the forward pass will have minimum error .So, the output of each neuron will get closer to its Target this is called the reverse pass. This process is repeated again and again to minimize the error.[15]

The Character is recognized after training the network with various datasets of that particular character to get maximum accuracy and minimum error.

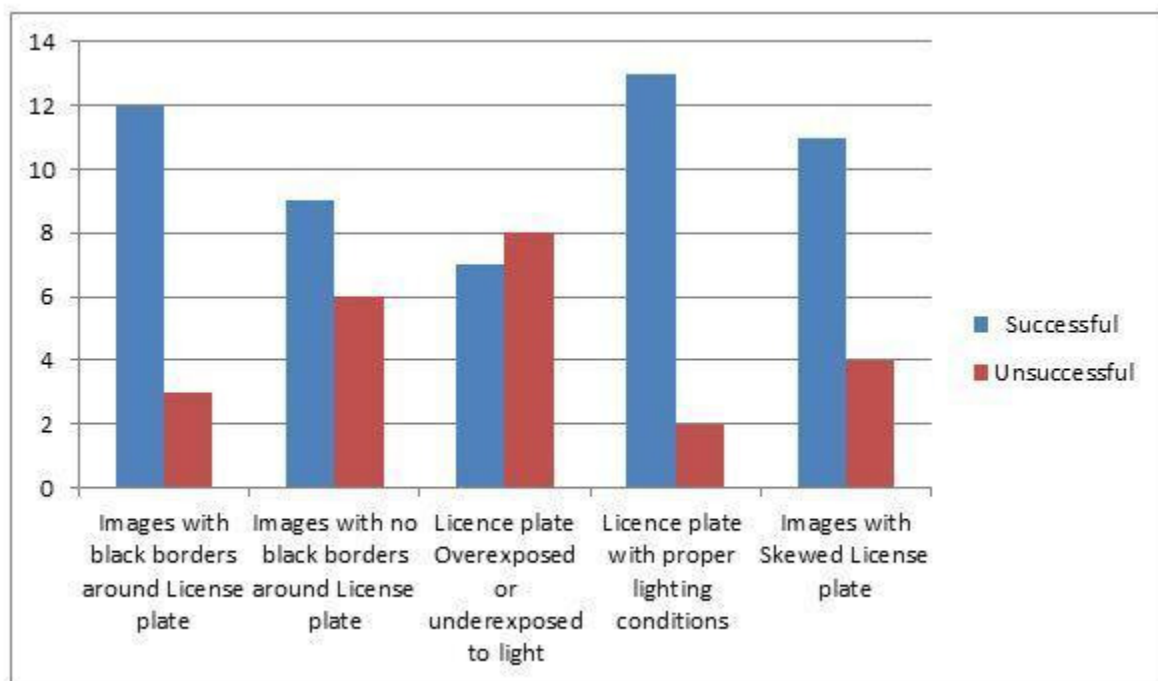
4 Applications

1. Automatic ticketing of vehicles at car parking facilities.
2. Tracking vehicles during traffic signal violations and related applications with huge saving of human energy and cost.

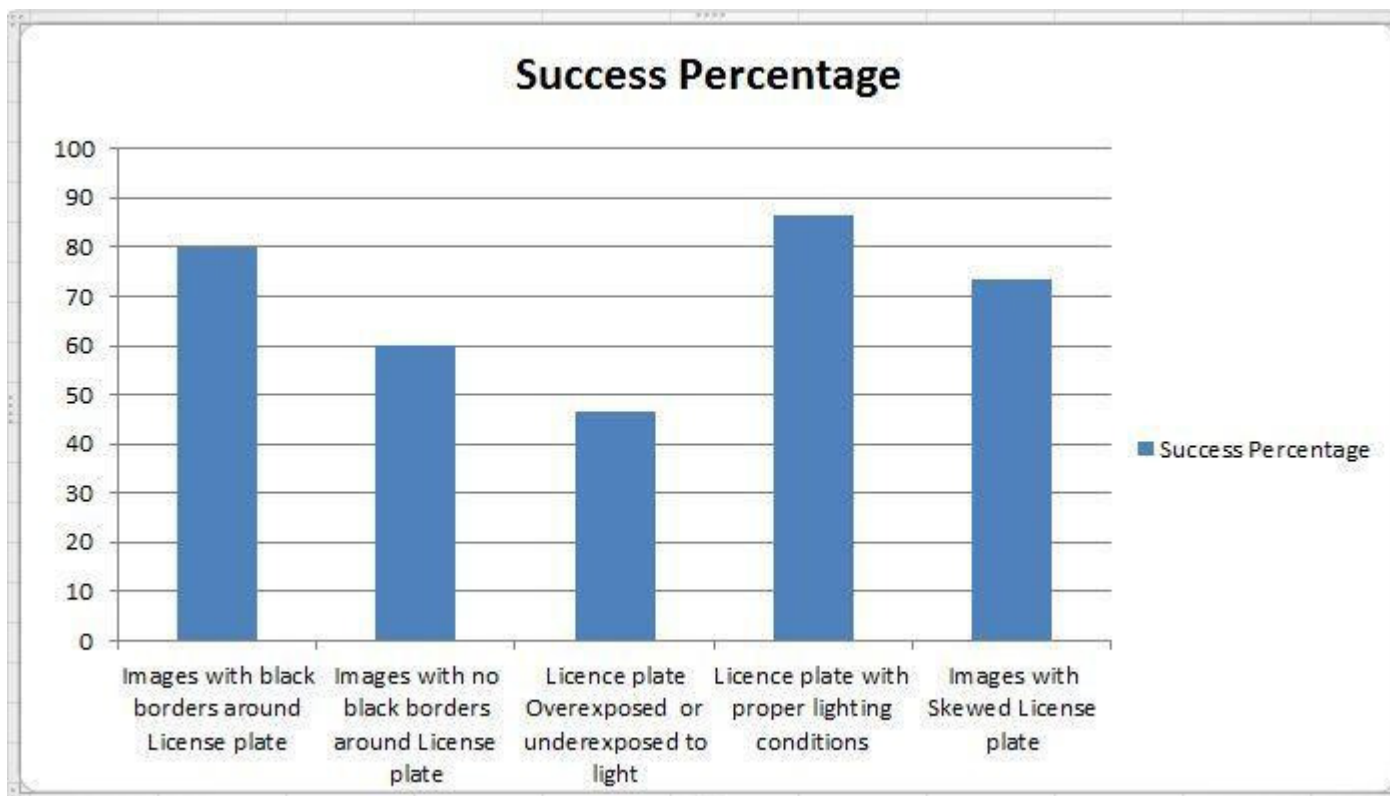
5 Results

The results obtained are as follows:

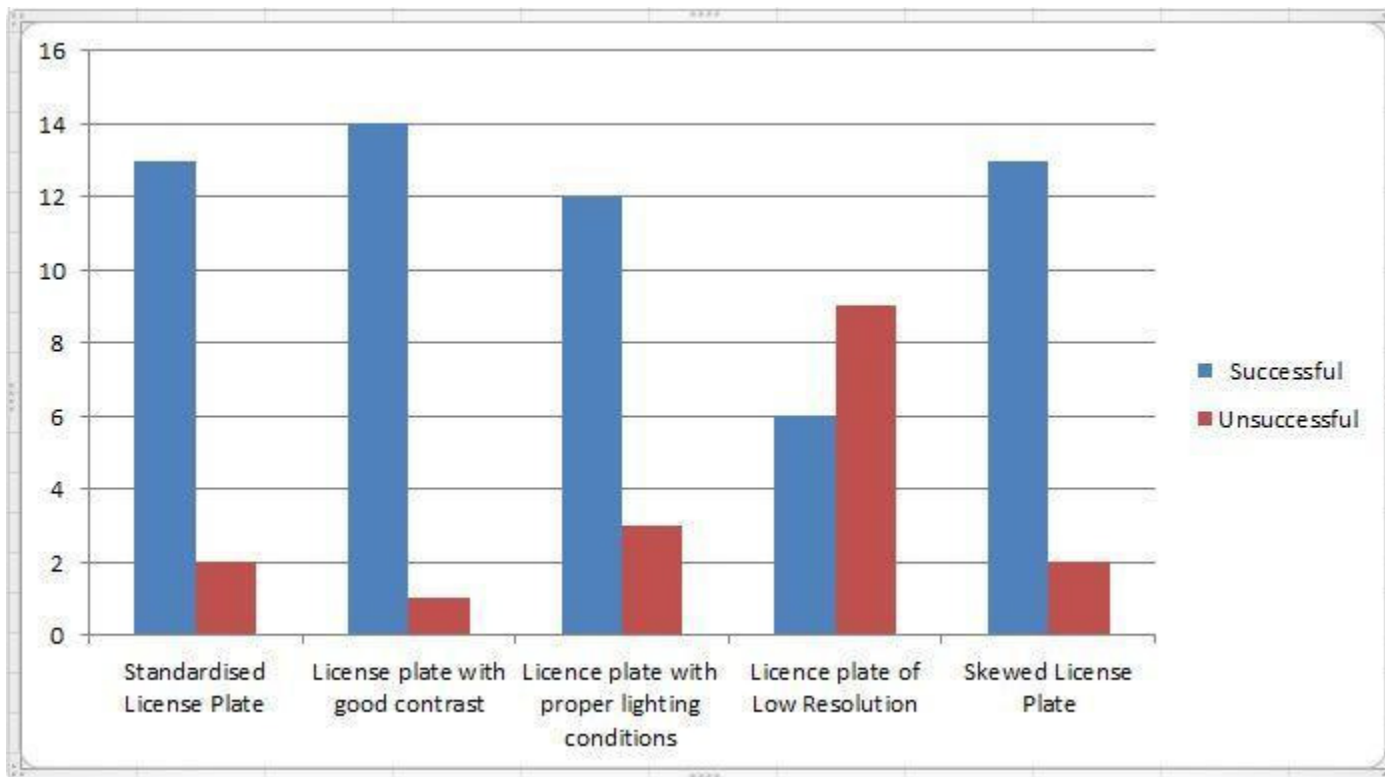
For the Module Localization of the License plate:



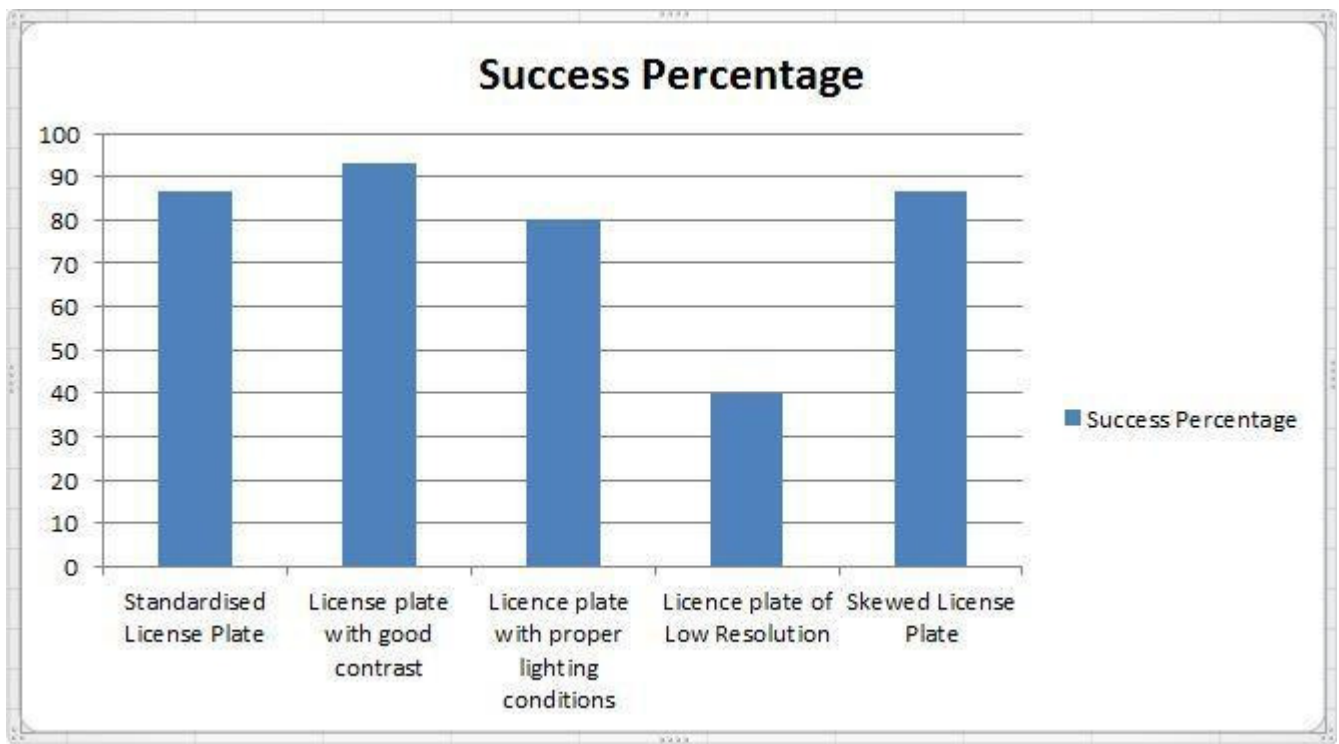
For the Module Localization of the License plate:



For the Module Segmentation of Localized of the License plate:



For the Module Segmentation of Localized of the License plate:



6 Conclusion

- For the localization of the license plate region there should be a proper edge in between the license plate boundary and the car in the background for the sobel operator to detect the edge .The camera should be present at a particular distance from the license plate so that the range in which the total number of pixels lie inside the license plate region remain constant.
- For the process of character segmentation we observed that if there is no clear boundary/peaks between each character, segmentation cannot be carried out successfully. Also fancy fonts create a hindrance to successful segmentation.

References:

- [1] Wisam Al Faqheri and Syamsiah Mashohor, "A Real-Time Malaysian Automatic License Plate Recognition (M-ALPR) using Hybrid Fuzzy" ,IJCSNS International Journal of Computer Science and Network Security, VOL.9 No.2, February 2009.
- [2] Saeed Rastegar, Reza Ghaderi, Gholamreza Ardeshir & Nima Asadi, " An intelligent control system using an efficient License Plate Location and Recognition Approach", International Journal of Image Processing (IJIP) Volume(3), Issue(5) 252, 2009.
- [3] V. Ganapathy and W.L.D. Lui, "A Malaysian Vehicle License Plate Localization and Recognition System", Journal of Systemics, Cybernetics and Informatics, Vol. 6, No. 1, 2008.
- [4] 3D surface tracking and approximation using Gabor filters, Jesper Juul Henriksen, South Denmark University, March 28, 2007
- [5] O. Martinsky, "Algorithmic and Mathematical Principles of Automatic Number Plate Recognition System", B. Sc. Thesis, BRNO University of Technology, 2007.
- [6] R. C. Gonzalez and R. E. Woods, Digital Image Processing, Pearson Education Asia, 2002.
- [7] Satadal Saha¹, Subhadip Basu, Mita Nasipuri, Dipak Kumar Basu," License Plate Localization from Vehicle Images:An Edge Based Multi-stage Approach", International Journal of Recent Trends in Engineering, Vol 1, No. 1, May 2009
- [8] Satadal Saha, Subhadip Basu, Mita Nasipuri, Dipak Kumar Basu "An Offline Technique for Localization of License Plates for Indian Commercial Vehicles"
- [9] Fukunaga K.: Introduction to statistical pattern recognition, Academic Press, San Diego, USA, 1990
- [10] Gonzalez R., Woods R.: Digital Image Processing, Prentice Hall, Upper Saddle River, New Jersey, 2002
- [11] "Edge Detection Tutorial" <http://www.pages.drexel.edu/~weg22/edge.html>
- [12]" Pixel connectivity-wikipedia" http://en.wikipedia.org/wiki/Pixel_connectivity
- [13] "Minimum bounding box-wikipedia"http://en.wikipedia.org/wiki/Minimum_bounding_box
- [14] "ai-junkie" <http://www.ai-junkie.com/ann/evolved/nnt2.html>
- [15]"The Back Propagation Algorithm" <http://www4.rgu.ac.uk/files/chapter3%20-%20bp.pdf>